OPEN ACCESS

University of BRISTOL

Dong, J., Yuan, J., Li, L., Zhong, X., & Liu, W. (2020). An Efficient Semantic Segmentation Method using Pyramid ShuffleNet V2 with Vortex Pooling. In *31st International Conference on Tools with Artificial Intelligence (ICTAI2019)* Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/ICTAI.2019.00-98

Peer reviewed version

Link to published version (if available):
10.1109/ICTAI.2019.00-98

Link to publication record in Explore Bristol Research
PDF-document

This is the author accepted manuscript (AAM). The final published version (version of record) is available online via IEEE at https://ieeexplore.ieee.org/document/8995234 . Please refer to any applicable terms of use of the publisher.

University of Bristol - Explore Bristol Research
General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/

An Efficient Semantic Segmentation Method using Pyramid ShuffleNet V2 with Vortex Pooling

Jiansheng Dong, Jingling Yuan*, Lin Li, Xian Zhong
School of computer science and technology
Wuhan University of Technology
Wuhan, China
1210090743@qq.com, {yjl, cathylilin,
zhongx}@whut.edu.cn

Weiru Liu
School of Computer Science, Electrical and Electronic Engineering, and Engineering Maths
University of Bristol
Bristol, UK
weiru.liu@bristol.ac.uk

***Abstract*—Efficient and accurate semantic segmentation is particularly important especially for applications like autonomous driving which requires real-time inference speed and high performance. Many works try to compromise spatial resolution to achieve real-time inference speed, which leads to poor performance. As a result, real-time segmentation task for embedded devices is still an open problem. In this paper, we focus on building a network with better performance possible while still achieve real-time inference speed. We first use a pyramid kernel size to capture more spatial information instead of using just a 3×3 kernel size for DWConvolution in ShuffleNet v2. Meanwhile, an efficient Vortex Pooling module is employed to aggregate the contextual information and generate high-resolution features. Compared with other state-of-the-art real-time semantic segmentation networks, the proposed network achieves similar inference speed and better performance on embedded device. Specifically, we achieve state-of-the-art 73.46% mean IoU on Cityscapes test dataset, for a 768×1024 input, a speed of 46.1 frames per second on NVIDIA Jetson AGX Xavier embedded development board is achieved.**

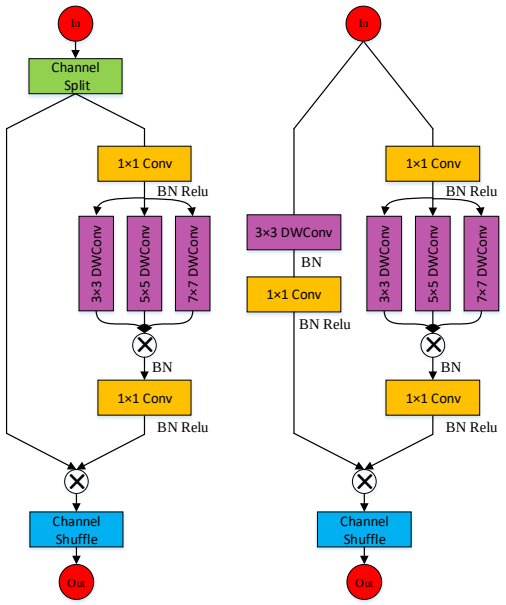
Keywords- semantic segmentation, real-time, embedded

I. INTRODUCTION

The research of semantic segmentation is a challenge in computer vision. Recent interest in autonomous driving, video surveillance, and medical image research has emerged a great demand for semantic segmentation algorithms that can operate in real-time on low-powerful embedded devices. Moreover, such as automatic driving, which requires fine accuracy, puts forward high requirements for the performance of semantic segmentation. Consequently, the semantic segmentation algorithms should be compact and computationally efficient. The lightweight neural network works should balance efficiency and accuracy.

Recently, many research works focus on accelerating semantic segmentation network to achieve low latency. These works can be summarized into three kinds of approaches that

- Restrict the input size to reduce the computation complexity by cropping or resizing [24, 31]. This is a simple and effective way to achieve high efficiency, but also lost much spatial information, resulting in poor performance in both metrics and visualization.



(a) Basic unit (b) Spatial down sampling unit

Figure 1. Pyramid building blocks. (a) Basic unit; (b) Spatial down sampling unit (stride = 2). DWConv: depthwise convolution. ⊗: concatenation.

- Try to compress the existing network by using knowledge distillation [8], pruning [7] and other compression algorithms. Its goal is to minimize the size of the network as much as possible, which will greatly alter or even destroy the original network structure.
- Design efficient architectures [11, 12]. This approach mainly uses Depthwise Separable Convolution (DWConvolution), which decomposes a convolution operation into Depthwise Convolution and Pointwise Convolution, greatly reduce computational requirements without significantly reducing accuracy.

Specifically in the third approach, ShuffleNets [13, 29] and other lightweight architectures [4, 9, 21] have designed for mobile devices based on DWConvolution show that this operation can effectively achieve the appropriate results with

fewer parameters. These lightweight architectures promise low-latency inference speed in the task of semantic segmentation and motivate us to explore efficient networks with rich spatial information.

In this paper, we introduce a lightweight semantic segmentation network that can achieve real-time inference on embedded devices with state-of-the-art performance. Our network first uses multi-scale kernels (a pyramid kernel size) to capture multi-level spatial information instead of using just a 3×3 kernel size for DWConvolution in ShuffleNet V2 [13]. Meanwhile, an efficient Vortex Pooling [25] module is employed to aggregate the contextual information and generate high-resolution features. Our main contributions are summarized as follows:

- We achieve computationally efficient inference on embedded devices, especially achieving 73.46% mIoU on Cityscapes test set using Pyramid ShuffleNet and an efficient Vortex Pooling.
- Proposed Pyramid ShuffleNet can effectively extract the multi-levels feature of the image and ensure the richness of spatial information. We accelerate Vortex Pooling to make it more efficient to aggregate the contextual information and generate high-resolution features.
- Proposed network is capable of running real-time on embedded devices.

II. Related Work

In this section, we introduce the state-of-the-art research in the task of semantic segmentation by analyzing the evolution process of semantic segmentation network.

Convolutional neural networks (CNNs) [10] can not only achieve state-of-the-art performance in the task of image classification, but also make great achievement in semantic segmentation.

Initially, image block classification is a deep learning method commonly used in semantic segmentation tasks, which uses the image blocks around each pixel to divide each pixel into corresponding categories. The main reason for using image blocks is that the classification network usually has a full connection layer, and its input needs to be a fixed size image block. Especially, the proposed seminal fully convolution network (FCN) [12] extends the original CNNs and can make an intensive prediction without full connection layer, which laid the foundation for most modern segmentation architectures.

In addition to the full connection layer structure, the pooling layer is another limitation that makes it challenging to use CNNs in segmentation problems. The pooling layer (down sampling) not only enlarges the receptive field of the upper convolution filter, but also aggregates the spatial information and discards part of the spatial information. However, the semantic segmentation method needs to adjust the category map accurately, so it requires both rich spatial information and sizeable receptive field. Researchers have proposed three different structures to solve the problem of spatial information loss.

Researchers have proposed three different structures to solve the problem of spatial information loss.

Encoder-Decoder Architecture: The encoder uses the pooling layer to reduce the spatial dimension of the input data gradually, while the decoder progressively restores the details and the spatial dimension of the target through the deconvolution layer and other network layers. There is usually a direct information connection between the encoder and the decoder to help the decoder recover the target details better. Both FCN [12] and SegNet [1] are early encoder-decoder structures, but the benchmark scores of SegNet cannot satisfy the practical requirements. Enet [17] also designed an encoder-decoder structure with few layers to reduce computational cost.

Some methods employ their specific refinement structure into U-shape [1, 6, 12, 16, 20] structure. Vijay et al. and Hyeonwoo et al. [1, 16] create an U-shape network with the usage of deconvolution layers. UNet [20] acquires multi-level features of input through skip-connected. Poudel et al. [6] proposed a feature fusion method inspired by Laplacian pyramids. Lin et al. [11] fuses coarse-grained high-level features and fine-grained low-level features. However, in the U-shape structure, some lost spatial information can not be easily recovered.

Two-branch Architecture: This Architecture usually has two branches to confront with the loss of spatial information and the contraction of receiving field respectively. ICNet [31], ContextNet [18], BiSeNet [26] and GUN [15] employ a shallow network structure as the spatial branch to obtain rich spatial information from low-level features. In respect of context branch, a deep network structure is employed to obtain a larger receptive field to acquire global context information. More recently, inspired by two-branch architecture, Fast-SCNN [19] incorporates a shared shallow network path to encode detail, while the context is efficiently learned at low resolution.

Nevertheless, networks with the two-branch or multi-branch architecture usually introduce heavy computational overhead, being nontrivial to optimize, especially when the network goes more in-depth, which therefore makes them unfavorable for the task of semantic segmentation.

Atrous Convolutions based Architecture: Atrous convolutions, or dilated convolutions [27], are shown to be a powerful tool in the semantic segmentation task [2]. The atrous convolutions amplify the receptive field of the convolution filter while keeping the number of parameters invariant. Simultaneously, it can promise that the size of the feature map remains invariant.

By using atrous convolutions it is possible to use pretrained ImageNet networks such as [13, 21] to extract denser feature maps by replacing downscaling at the last layers with atrous rates, thus allowing us to control the dimensions of the features [22]. DeepLab V3 [2] is one of the most recent state-of-the-art semantic segmentation networks on multiple benchmarks. In their approach, they improved the ASPP module proposed in [3] for better context features. Furthermore, Vortex Pooling [25] delves into the ASPP module and explore its deficiency. Meanwhile, Yu et al. [28] and Wang et al. [23] showed that dilated convolution might

cause "gridding" problems, and they proposed Hybrid Dilated Convolution (HDC) to remove such abnormal artifacts.

Our literature review in this Section and Section 1 shows us that the architecture and implementation of ShuffleNet V2 are effective as the main feature extraction structure of efficient semantics segmentation network. Both Türkmen et al. [22] and Zhao et al. [30] indicate that employing ShuffleNet can achieve fast inference speed and acceptable accuracy. We improve ShuffleNet V2 to capture multi-level spatial information and accelerate Vortex Pooling to aggregate the contextual information and generate high-resolution features.

III. PROPOSED METHOD

In this section, we first illustrate our proposed pyramid building blocks. Furthermore, we describe improved Vortex Pooling. Finally, we elaborate our proposed network architecture.

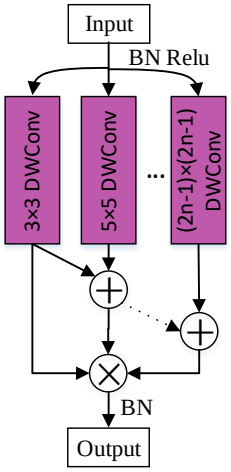


Figure 2. The combination strategy of output group of the pyramid convolution. ⊕: addition.

A. Pyramid Building Blocks

As shown in Figure 1, we use a pyramid kernel size to capture more spatial information instead of using just a 3×3 kernel size for DWConvolution in building blocks of ShuffleNet v2 [13]. Then combines all output of pyramid convolution before the 1×1 convolution.

For basic unit, the "Channel Split" operation splits the input into two branches with c-c' and c' channels, respectively. For simplicity, we set $c'=c/2$. One branch directly goes through the block without any operation. The other branch consists of a pyramid convolution sandwiched between two 1×1 convolutions with the same input and output channels. After convolution, the two branches are concatenated. Finally, the "Channel shuffle" operation is employed to enable information communication between the two branches.

For spatial down sampling, The "Channel Split" operation is removed at the beginning of the block. Thus, the number of output channels is doubled. Besides, the left branch consists of a 1×1 DWConv (stride = 2) and a 1×1 convolutions. The right branch remains the same as the basic unit except that the stride of convolutions whose size larger than 1 is set to 2.

Furthermore, we set the kernel size group of pyramid convolution to $K=\{k_1,k_2,\ldots,k_N\}$. We apply a strategy (as shown in Figure 2) inspired by [5] to combine the outputs of pyramid convolution. For the output group $C=\{c_1,c_2,\ldots,c_N\}$ of the pyramid convolution, the final combined output is

$$output=\prod_{i=1}^{N}\sum_{j=1}^{i} c_j \tag{1}$$

where $\prod_{i=1}^{N} x_i$ means concatenating $x_1, x_2, \ldots, x_N$, and $\sum_{i=1}^{N} x_i$ means adding $x_1,x_2,\ldots,x_N$.

B. Efficient Vortex Pooling

As aforementioned, semantic segmentation requires both rich spatial information and sizeable receptive field. In other words, besides spatial information, global contextual information is also essential. Vortex Pooling is an effective module for aggregating contextual information by multi-branch convolution with different dilation rates. Different dilation rates can dramatically increase the receptive field, thus acquiring multi-level contextual information.

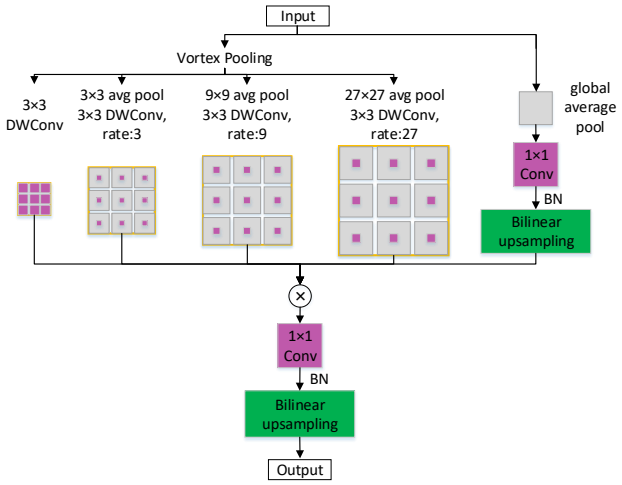


Figure 3. Architecture of efficient Vortex Pooling.

Vortex Pooling first takes each k×k square region in input feature map as subregion. Specifically, it uses small k for the subregions near from the given pixel, which enables more details. While for regions far away from the target pixel, it uses large k because only contextual information is needed. The k values for the four convolution layers are set to (1, 3, 9, 27), respectively. Operationally, it first uses k×k average pooling to pool the descriptor in each subregion to one new descriptor. Then it employs four convolution layers with different dilation rates to aggregate the descriptors from the

TABLE I. RESULTS ON CITYSCAPES TEST SET OF (1) ENET [17], (2) SHUFFLENETV2+DPC [22], (3) FAST-SCNN [19], AND OUR PYRAMID SHUFFLENET V2 WITH EFFICIENT VORTEX POOLING.

Method	mIoU	Building	Sky	Car	Sign	Road	Person	Fence	Pole	Sidewalk	Bicycle
1	58.3	85.0	90.6	90.6	44.0	96.3	65.5	33.2	43.5	74.2	55.4
2	70.3	90.7	93.9	94.0	66.9	98.1	78.5	50.9	51.5	82.5	67.5
3	68.0	89.7	94.3	93.0	60.5	97.9	74.0	48.6	48.3	81.6	61.2
Ours	**73.4**	**91.6**	**94.6**	**94.7**	**73.8**	**98.3**	**82.4**	**51.6**	**58.2**	**83.8**	**70.8**

all subregions. Simultaneously, it applies global average pooling on the input to incorporate global information, feeds the result to a 1×1 convolution, and then bilinearly up sample the feature map to generate high-resolution features.

TABLE II. PROPOSED NETWORK ARCHITECTURE.

Layer	Output Size	Stride	Rate	Repeat
Image	512×1024×3			
Conv2D	256×512×24	2		1
MaxPool	128×256×24	2		
Stage1	64×128×116	2	1	1
	64×128×116	1	1	3
Stage2	32×64×232	2	1	1
	32×64×232	1	1	7
Stage3	32×64×464	1	1	1
	32×64×464	1	(1, 2, 3)	3
Vortex Pool	128×256×256			1
Conv2D	128×256×128	1		1
Conv2D	128×256×n_classes	1		
Bilinear Up	512×1024×n_classes			1

While the original Vortex Pooling effectively converges contextual information, it also introduces a lot of additional parameters from the four convolution layers. This stunts the inference speed of our network heavily. To address the problem, we apply four 3×3 depthwise separable convolutions with dilation rates are set to (1, 3, 9, 27) instead of standard convolutions as shown in Figure 3. This operation remarkably reduces the number of parameters and improves efficiency, while almost maintaining the original performance of Vortex Pooling. To be precise, it reduces the number of parameters from 10.42 million to 2.69 million at the expense of 0.7% mIoU (as shown in Table IV).

C. Network Architecture

The architecture of our proposed network is presented in Table II. It can be divided into an initial module, a feature extraction module, Vortex Pooling module, and up-sampling module.

The initial module consists of a standard convolution and a max pooling, and feature extraction module consists of three stages. These two modules are based on ShuffleNet v2. Note that there are some differences. For following the discovery in [2], that is, the consecutive striding is harmful for semantic segmentation. We set the stride of the spatial down sampling unit in Stage3 to 1, the dilated rates of the next three basic units to (1, 2, 3) respectively, and the kernel size of all DWConvs of Stage3 to 3×3. In the original ShuffleNet v2, *output_stride* goes as low as 32, after our implementation, the output_stride of Stage3 in our network is adjusted to 16.

Next module is the improved efficient vortex pooling as previously described. Finally, two convolutions and a bilinear up sampling are employed to classify and up sample the feature map to input size.

IV. EXPERIMENTS

We evaluate our proposed network on Cityscapes [14] benchmark, which is a large urban street scene dataset in the field of semantic segmentation. It contains 5000 finely annotated samples which are split into 2975, 500 and 1525 for training, validation, and testing respectively.

A. Implementation Details

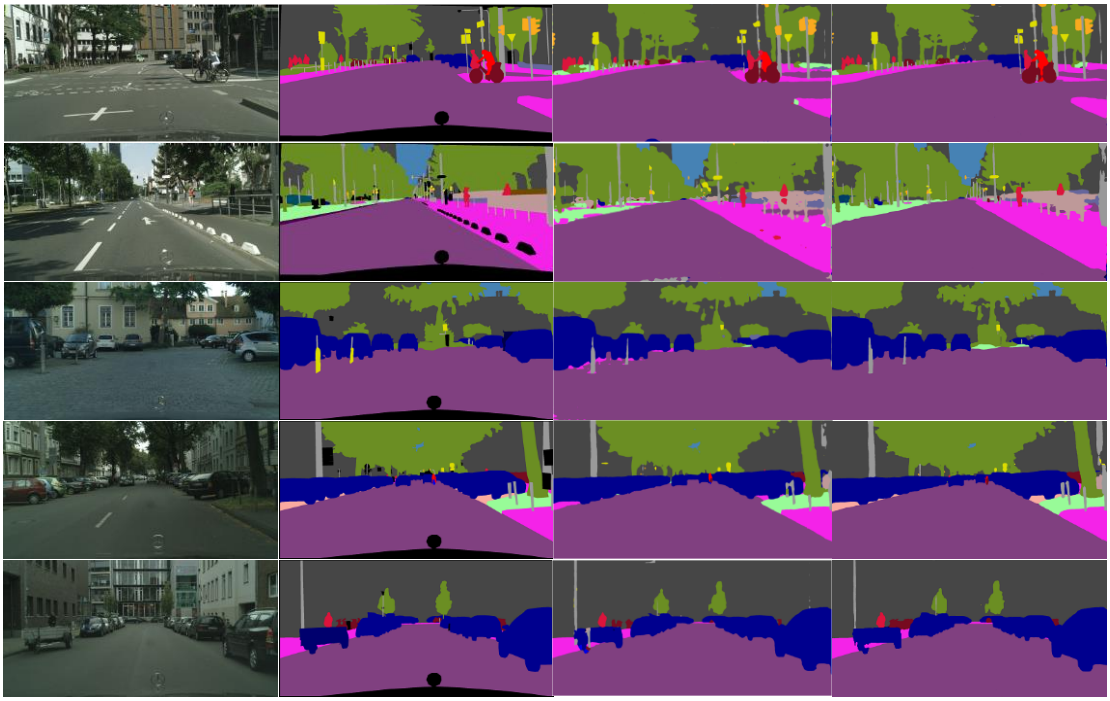
We use stochastic gradient descent (SGD) with momentum 0.9 and batch-size 16. Following [9, 14, 19, 30], we employ "ploy" learning rate with initial value $2.5e^{-2}$ and power 0.9. Our network is trained with cross-entropy loss. To augment data, we apply random resizing with scales contains {0.75, 1.0, 1.5, 1.75, 2.0}, randomly crop, horizontal flip.

B. Evaluation of Cityscapes

TABLE III. CLASS AND CATEGORY IOUS OF OUR PROPOSED NETWORK COMPARED TO OTHER STATE-OF-THE-ART SEMANTIC SEGMENTATION NETWORKS ON THE CITYSCAPES TEST SET. THE NUMBER OF PARAMETERS IS LISTED IN MILLIONS.

Mothod	Class IoU	Cat. IoU	Params
DeepLab-v2 [3]	70.4	86.4	**44. __**
PSPNet [30]	**78.4**	**90.6**	65.7_
SegNet [1]	56.1	79.8	29.46
Enet [17]	58.3	80.4	**00.37**
ICNet [31]	69.5	-	06.68
ERFNet [32]	68.0	86.5	02.1_
BiSeNet [26]	71.4	-	05.8_
ShuffleNetv2+DPC [22]	70.33	86.48	03.82
Fast-SCNN [19]	68.0	84.7	01.11
Ours	**73.46**	**88.32**	02.69

Our network outperforms state-of-the-art efficient networks on the Cityscapes test set. Table I display detail results of IoUs on class-level. As shown in Table III, we compare our network with other offline networks (DeepLab-v2, PSPNet) and state-of-the-art real-time semantic segmentation networks in terms of performance and number of parameters. Our network only has 2.69 million parameters, which is slightly higher than ERFNet (2.1_ million) and Fast-SCNN (1.11 million) except ENet with poor performance. Furthermore, We achieve 73.4% mIoU which makes a gain of 3.13% mIoU over the best performing ShuffleNetv2+DPC network.



(a) image (b) Ground truth (c) ShuffleNetv2+DPC (d) ours

Figure 4. Example results on Cityscapes validation set. (Black colored regions on ground truth are ignored)

C. *Ablation Experiments*

In this section, using mIoU and number of parameters as indicators, we perform ablation experiments to explore the effects of different implementation of the network.

TABLE IV. ABLATION RESULTS ON THE CITYSCAPES TEST SET. NUMBER OF PARAMETERS IS LISTED IN MILLIONS.

Method	Class	Params
ShuffleNetv2	67.7	**01.68**
PydShuffleNetv2	71.22	02.10
PydShuffleNetv2+VortexPooling	**74.16**	10.42
PydShuffleNetv2+efficirnt VP	73.46	02.69

The ablative results are showed in Table IV. Original ShuffleNet v2 can obtain 67.7% mean IoU on the Cityscapes test set. After we use pyramid convolutions instead of DWConvolution in ShuffleNet v2, mIoU increased by about 3.52%. Then Vortex Pooling further increases about 2.94% mIoU. For the number of parameters, as aforementioned, original Vortex Pooling introduces 8.3 million parameters from the four convolution layers, but our operation remarkably reduces the number of parameters from 10.42 million to 2.69 million. The results of ablation experiments strongly demonstrate the effectiveness of our continuous improvement of the network.

D. *Inference Speed on Embedded Devices and Qualitative Results*

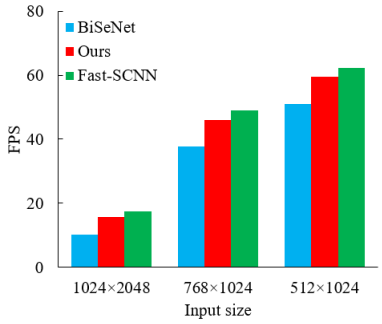


Figure 5. Inference speed on Jetson AGX Xavier.

We present the inference speed results for different input size on Jetson AGX Xavier Developer Kit in Figure 5. The configuration of Xavier is 512-core Volta GPU with Tensor Cores, 8-core ARM v8.2 64-bit CPU with 8MB L2 + 4MB L3 and 16-GB memory. The original image size of Cityscapes is 1024×2048, we also experiment with 768×1024 and 512×1024 resolutions. The experimental results show that our network achieves similar inference speed and better performance compared (over 2.06% and 5.46% mIoU of BiSeNet and Fast-SCNN respectively) with other state-of-the-

art real-time semantic segmentation networks on embedding device.

Finally, Figure 4 displays the qualitative results of our network on Cityscapes validation set. Visually, our network is closer to ground truth in some details. For example, the fence in the second picture, the pedestrian in the far of the fourth picture, and the tricycle in the fifth picture. In particular, our results have fewer noise points.

V. Conclusions

In this paper, we have presented a semantic segmentation network with state-of-the-art mIoU without compromising the inference speed. The experimental results show that our network achieves a competitive 73.46% mIoU on Cityscapes test dataset. Furthermore, we tested our network on embedded device and achieve 46.1 frames per second for a 768×1024 input, which means that our proposed network is capable of running real-time on embedded devices. Future work is to further accelerate the network without compressing performance.

From the experimental results, we discovered that the precise segmentation of object boundary is a direction in which the network can be improved. In the future work, we intend to combine some edge segmentation algorithms with our network efficiently, so as to further improve the performance of the network.

References

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12):2481–2495, 2017.

[2] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv: 1706.05587, 2017.

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Parsing NetworkICNet: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence, 40(4):834–848, 2018.

[4] Fran¸cois Chollet. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1251–1258, 2017.

[5] Mostafa Gamal, Mennatullah Siam, and Moemen Abdel-Razek. Shuffleseg: Real-time semantic segmentation network. arXiv preprint arXiv:1803.03816, 2018.

[6] Golnaz Ghiasi and Charless C Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In European Conference on Computer Vision, pages 519–534. Springer, 2016.

[7] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149, 2015.

[8] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531, 2015.

[9] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko,WeijunWang, TobiasWeyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012.

[11] G Lin, A Milan, C Shen, and I Reid. Refinenet: Multi-path refinement networks with identity mappings for high-resolution semantic segmentation. arxiv 2016. arXiv preprint arXiv:1611.06612.

[12] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3431–3440, 2015.

[13] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), pages 116–131, 2018.

[14] Sebastian Ramos Timo Rehfeld Markus Enzweiler Rodrigo Benenson Uwe Franke Stefan Roth Bernt Schiele Marius Cordts, Mohamed Omran. The cityscapes dataset for semantic urban scene understanding. IEEE conference on computer vision and pattern recognition, 2018.

[15] Davide Mazzini. Guided upsampling network for real-time semantic segmentation. arXiv preprint arXiv:1807.07466, 2018.

[16] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE international conference on computer vision, pages 1520–1528, 2015.

[17] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. arXiv preprint arXiv:1606.02147, 2016.

[18] Rudra PK Poudel, Ujwal Bonde, Stephan Liwicki, and Christopher Zach. Contextnet: Exploring context and detail for semantic segmentation in real-time. arXiv preprint arXiv:1805.04554, 2018.

[19] Rudra PK Poudel, Stephan Liwicki, and Roberto Cipolla. Fast-scnn: Fast semantic segmentation network. arXiv preprint arXiv:1902.04502, 2019.

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[21] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4510–4520, 2018.

[22] Sercan Türkmen and Janne Heikkilä. An efficient solution for semantic segmentation: Shufflenet v2 with atrous separable convolutions. arXiv preprint arXiv:1902.07476, 2019.

[23] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell. Understanding convolution for semantic segmentation. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 1451–1460. IEEE, 2018.

[24] Zifeng Wu, Chunhua Shen, and Anton van den Hengel. Real-time semantic image segmentation via spatial sparsity. arXiv preprint arXiv:1712.00213, 2017.

[25] Chen-Wei Xie, Hong-Yu Zhou, and Jianxin Wu. Vortex pooling: Improving context representation in semantic segmentation. arXiv preprint arXiv:1804.06242, 2018.

[26] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), pages 325–341, 2018.

[27] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122, 2015.

[28] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 472–480, 2017.

[29] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6848–6856, 2018.

[30] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, XiaogangWang, and Jiaya Jia. Pyramid scene parsing network. 2016.

[31] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images. In Proceedings of the European Conference on Computer Vision (ECCV), pages 405–420, 2018.

[32] E. Romera, J. M. A´lvarez, L. M. Bergasa, and R. Arroyo. ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation. IEEE Transactions on Intelligent Transportation Systems, 19(1): 263-272, 2017.